

Enero del 2019

Microseguros

Logger with Serilog

	Owner	
Diego O. Miguel	Team Leader	domiguel@sofrecom.com.ar
Emiliano G. Santellan	Team Scrum	esantellan@sofrecom.com.ar
Edward E. Schmoll	Team Scrum	eschmoll@sofrecom.com.ar

Responsable: Edward E. Schmoll

Revisión: Diego O. Miguel

Configuración	2
Utilización	3
Modo de uso	3
Recuperación de Registros	4

Configuración

La configuración se extrajo del siguiente [link](#).

Utilización

[Serilog](#) es un servicio para registrar los eventos de las transacciones que genera la aplicación. En el siguiente [link](#) se pueden ver los distintos eventos que brinda.

Modo de uso

En el servicio donde usted quiere agregar *Serilog* se recomienda declarar una variable privada de solo lectura de la siguiente forma (el siguiente ejemplo se tomó de un extracto de la clase *DeviceService*):

```
...  
private readonly ILogger<DeviceService> _logger;  
...
```

Luego se pide el servicio en el constructor y el servicio de inyección de dependencias nos pasará la instancia como se muestra en el siguiente ejemplo:

```
public DeviceService(ILogger<DeviceService> logger,...)  
{  
    _logger = logger;  
    ...  
}
```

Luego ya está listo para usar en los distintos métodos con los que cuenta dicha clase, en el siguiente ejemplo se muestra cada registro en el método que devuelve todos los dispositivos:

```
public IEnumerable<Device> GetAll()  
{  
    try  
    {  
        _logger.LogInformation("Recuperando todos los dispositivos");  
        return _deviceRepository.Get();  
    }  
    catch (Exception ex)  
    {  
        _logger.LogError(ex, "Error al obtener los dispositivos");  
        throw ex;  
    }  
}
```

Recuperación de Registros

Serilog se configuró para que persista la información en la tabla de Log en la base que se configuró en appsettings.json.

Por ejemplo se quiere visualizar los errores que ocurrieron en los últimos 30 días:

```
select * from log
where timestamp >= GETUTCDATE() - 30
and level = 'Error'
order by timestamp desc
```